

```

--Segunda Chamada da Prova de Paradigmas de Linguagens Computacionais
-- 2/2017 - 14/12/2017
--
-- Nome:

-- 1) (2.5) Escreva uma função f :: [Int] -> [Int] -> Int que compute a soma
dos números no seu primeiro argumento que são divisíveis pelo número na sua
posição correspondente no segundo argumento. Se os comprimentos das listas não
forem os mesmos, os elementos extras na lista mais longa devem ser ignorados.
Assuma que nenhum dos números na segunda lista é zero. Por exemplo:

-- f [6,9,2,7] [2,3,5,1]      = 22
-- f [6,9,2] [2,3,5,1]        = 15
-- f [1,2,3,4,5] [5,4,3,2,1]  = 12
-- f [10,20,30,40] [3,4,5,6,7] = 50

-- 2) (2.5) Escreva uma função p :: String -> Int que retorne o maior dígito da
string, retornando zero se a string não tiver dígitos. Por exemplo:

-- p "Inf1-FP"      = 1
-- p "Functional"    = 0
-- p "1+1=2"         = 2
-- p "3.157/3 > 19" = 9

-- 3) Um robô é controlado por 2 comandos e se move em uma linha infinita como
a abaixo:

-- (posição)      -3  -2  -1  0  1  2  3
--                ... ----- ...

-- seus comandos são:

data Command =
    Go Int -- move o robô na direção em que estiver
  | Turn   -- inverte a direção
  deriving Show

-- antes e depois de cada movimento o robô está em um "estado" - em uma posição
na linha e virado para a esquerda ou a direita:

type Position = Int
data Direction = L -- Left
               | R -- Right
               deriving Show
type State = (Position, Direction)

-- 3a) (2.5) Escreva uma função move :: Command -> State -> State que, dado um
comando e um estado do robô, retorna o estado do robô após o comando.
-- Por exemplo:
-- move (Go 3) (0,R) = (3,R)
-- move (Go 3) (0,L) = (-3,L)
-- move Turn (-2,L) = (-2,R)

-- 3b) (2.5) Escreva uma função multimove :: [Command] -> State -> State que,
dada uma lista de comandos e um estado do robô, retorna o estado final do robô
após executar todos os comandos.
-- Por exemplo:

-- multimove [] (3,R) = (3,R)
-- multimove [Go 3, Turn, Go 4] (0,L) = (1,R)
-- multimove [Go 3, Turn, Turn] (0,R) = (3,R)
-- multimove [Go 3, Turn, Go 2, Go 1, Turn, Go 4] (4,L) = (0,L)

```